

Remarks

Reconsideration of this application as amended is respectfully requested.

Claims 1-19 stand rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 6,134,583 of *Herriot* ("*Herriot*") and U.S. Patent No. 5,636,355 of *Ramakrishnan et al.* ("*Ramakrishnan*") and U.S. Patent No. 4,633,387 of *Hartung et al.* ("*Hartung*") and U.S. Patent No. 6,055,526 of *Ambroziak* ("*Ambroziak*").

Claims 1-19 have been cancelled. New claims 20-46 have been added.

Applicants respectfully submit that new claim 20 is not obvious in view of the references cited by the Examiner because none of the references cited by the Examiner teach or suggest a class loader that converts classes obtained via a network into arrays and references of a predefined class definition format for storage in a class structure as claimed in new claim 20.

Applicants further submit that new claim 20 is not obvious in view of the references cited by the Examiner because none of the references cited by the Examiner teach or suggest a memory manager that selects and purges the arrays and references from a class structure so as to minimize an amount of the memory consumed by the class structure and to minimize class loading activities on a network as claimed in claim 20.

The Examiner has stated that "*Ramakrishnan* discloses purging selected ones of the predefined classes (i.e. data) from the class structure (e.g. col 4, line 24-29)." (Page 3, Office Action mailed 9-27-02). Applicants respectfully submit, however, that *Ramakrishnan* does not disclose classes or a class structure as claimed in new claim 20. Instead, *Ramakrishnan* discloses a disk cache management system that purges data from a write cache and writes it to a disk rather than purge classes from a

class structure as claimed in claim 20. For example, the section of *Ramakrishnan* cited by the Examiner is as follows

preselected lower threshold; and means operative if the step of deciding whether to purge to the disk results in a decision to initiate purging, for selecting at least one block from the write cache memory, then waiting till the disk is not busy with read operations, and writing the selected block or blocks to the disk. The apparatus also performs a purging...

(*Ramakrishnan*, col. 4, lines 24-29) (emphasis added). In further contrast, *Ramakrishnan* discloses purging data from a write cache to a disk in a manner that minimizes disk access delays (*Ramakrishnan*, col. 3, lines 11-15) rather than purge classes from a class structure to minimize an amount of the memory consumed by the class structure and to minimize class loading activities on a network as claimed in claim 20.

The Examiner has stated that that "Ambroziak discloses minimize an amount of the memory consumed by the predefined classes in the class structure (e.g. col 8, line 53-62)." (Page 3, Office Action mailed 9-27-02).

Applicants respectfully submit, however, that *Ambroziak* does not disclose minimizing an amount of memory consumed by a class structure as claimed in new claim 20.

Instead, *Ambroziak* teaches minimizing the memory needed to store an information index for a search engine. For example, the section of *Ambroziak* cited by the Examiner is as follows

Another property used to achieve compression is the fact that the document index file, the documents file, and the position index file can all be reconstructed from the positions file. Thus, only one file, the positions file, needs to be transmitted to a distributed search engine, instead of four files (the document index file, the documents file, the positions index file, and the positions file). This reconstruction property thus serves to reduce the amount of information transmitted to transmit the indexing structure and

minimizes the memory needed to store the indexing structure.
(*Ambroziak*, col. 8, lines 53-62) (emphasis added). It is respectfully submitted that minimizing memory by compressing a file does not suggest minimizing an amount of memory consumed by a class structure by selecting and purging classes as claimed in new claim 20.

The Examiner has stated that that "*Hartung* discloses minimize class loading activities on the network (e.g. col 4, line 40-46)." (Page 3, Office Action mailed 9-27-02). Applicants respectfully submit, however, that *Hartung* does not disclose minimizing class loading activities on a network as claimed in new claim 20. Instead, *Hartung* teaches load balancing in a peripheral data storage hierarchy (*Hartung*, col 6, lines 1-6). For example, the section of *Hartung* cited by the Examiner is as follows

In a dynamic data processing system where activity can vary beyond a control of the controlling data processors, load balancing between the various data processors/data processing paths should fully accommodate subsequent unforeseen dynamic changes in activities such that data transfers for load balancing are minimized for maximizing data processing throughput.

(*Hartung*, col. 4, lines 40-46) (emphasis added). It is respectfully submitted that minimizing the data transfers used for load balancing among storage devices does not suggest minimizing class loading activities on a network as claimed in new claim 20.

It is also submitted that *Ramakrishnan* does not teach or suggest a combination with *Herriot* and that *Ambroziak* does not teach or suggest a combination with *Herriot* and that *Hartung* does not teach or suggest a combination with *Herriot* and that *Herriot* does not teach or suggest a combination with *Ramakrishnan* or *Ambroziak* or *Hartung*. It would be impermissible hindsight based on an applicant's own disclosure to incorporate the

teachings of *Ramakrishnan* and *Ambroziak* and *Hartung* into the teachings of *Herriot*. Moreover, any such combination would still lack a class loader that converts classes obtained via a network into arrays and references of a predefined class definition format for storage in a class structure as claimed in new claim 20.

It is therefore respectfully submitted that the virtual machine of new claim 20 that includes a class loader that converts classes obtained via a network into arrays and references of a predefined class definition format and that includes a memory manager that selects and purges the arrays and references is not obvious in view of the teachings of *Herriot* and *Ramakrishnan* and *Ambroziak* and *Hartung* cited by the Examiner.

Given that new claims 21-30 depend from new claim 20, it follows that new claims 21-30 are not obvious in view of the references cited by the Examiner.

It is also submitted that new claims 31 and 40 are not obvious in view of the references cited by the Examiner. New claims 31 and 40 include limitations similar to the limitations of new claim 20. Therefore, the remarks stated above with respect to new claim 20 also apply to new claims 31 and 40.

It is further submitted that new claims 32-39 and 41-46 are not obvious in view of the references cited by the Examiner. New claims 32-39 and 41-46 depend from new claims 31 and 40, respectively, which are not obvious in view of the references cited by the Examiner. Therefore, it follows that new claims 32-39 and 41-46 are not obvious in view of the references cited by the Examiner.

It is respectfully submitted that in view of the amendments and arguments set forth above, the applicable rejections have been overcome.

The Commissioner is authorized to charge any underpayment or credit any overpayment to Deposit Account No. 08-2025 for any matter in connection with this response, including any fee for extension of time, which may be required.

Respectfully submitted,

Date: 12-24-02

By: _____

Paul H. Horstmann
Paul H. Horstmann
Reg. No.: 36,167